

Cory Spitzer
The Linux Command Line
Jim Mahoney
May 3rd, 2013

The Linux Command Line Tutorial: Thoughts and Observations

This summer I started thinking that I might want to study Computer Science. Because of this, I decided I needed to either upgrade my then current laptop, or acquire a new one. I had previously used only poorly built legacy systems or proprietary unix-like systems, but when I thought of how to upgrade my computing resources, Linux was the obvious choice to explore as an alternative. I knew very little about Linux, but I did not that it was what all the cool kids used. I set about trying to figure out what it was all about, and which one to use.

My first mission was to get my 512MB IBM T40 to run (instead of traipsing along as if on a summer stroll in an English park). After figuring out the difference between different operating systems and different desktop environments, I started installing different light weight Linux distributions on it. Some were too slow, some were too hot, some just had too many angry bears¹. I finally got something reasonable to work, but decided that a new laptop was the way to go. I couldn't bear buying a new legacy machine, as that would be silly, nor an expensive unix-y machine, as that would be expensive. So, I ended up with a spiffy new Ubuntu machine. So it goes.

Of course, to *really* be cool, I would have to eschew the purty pangolin background and fancy-shmancy Dash and head for the deep depths of the all powerful CLI! I first learned to walk and look around it in “Intro to Programming,” so I was excited to explore. To the Linux tutorial! I picked out a couple of books and started going through the aptly named *The Linux Command Line*.

Essentially, I went through the chapters and followed along with the examples. By doing this I gained familiarity with what could be done from the command line. I only really remembered the tools

¹ Bearuntu, which specializes in honeypots, is currently in bearta development.

that I had reason to use repeatedly, but I gained enough familiarity with the ones I only used a few times to know that they exist, should I need them. I learned the basics of editing with vim, compiling programs from source, writing very basic bash scripts and piping: *ls | less*. I tried to setup an RSA key to login with *ssh* on remote computer. I covered and explored the following commands and some of their options:

ls, cd, pwd, man, apt-get
type, mkdir, mv, help, info, echo
sort, less, wc, grep, head, tail, alias
chmod, umask, sudo su
ps, top, jobs, kill
ssh, sftp, and wget
vim esc:wq!, i
gzip/gunzip/zless/zcat, bzip2/bunzip2
alt + SysRq + r, i, e, s, u, b - (reboot even if system utterly broken)²
gcc, make

What might I do different? I might find a resource with more exercises, so that I wasn't just following along with the examples. I might have covered a few less commands, but in more detail. Overall, I think I'm pretty happy with what I covered. Its easy to find various lessons and tutorials in books and online, but they don't provide the context that a tutorial does for situating oneself and having some idea of what territory one has explored.

2 Not necessarily done from the command line – talks directly to BIOS